

КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
Факультет географии и экологии

СОСТАВЛЕНИЕ В СУБД FOXPRO  
ПРОГРАММЫ ЗАПОЛНЕНИЯ БАЗЫ ДАННЫХ

Учебно-методическое пособие

КАЗАНЬ – 2007

## **Составитель**

Кандидат физико-математических наук, старший преподаватель  
Е.А.Костерина

Учебно-методическое пособие содержит в себе простой пример создания в СУБД FoxPro базы данных, состоящей из трех таблиц. Оно задумано как помощь студентам факультета географии и экологии при выполнении практических заданий в СУБД FoxPro.

Печатается по решению кафедры моделирования экологических систем Казанского государственного университета от 07.03.07 (Протокол № 6) и по решению учебно-методической комиссии факультета географии и экологии от 17.04.07 (Протокол № 4).

## ПОСТАНОВКА ЗАДАЧИ

Написать программу заполнения базы данных, состоящей из трех таблиц. Выбор действия (добавить, изменить, удалить запись) и выбор таблицы, в которой нужно совершить действие, должны осуществляться с помощью меню программы. Каждому действию в каждой таблице должна соответствовать отдельная процедура.

Таблица 1. Информация о студентах.

Код студента	Фамилия студента	Имя студента	Отчество студента	Номер группы
--------------	------------------	--------------	-------------------	--------------

Первичный ключ таблицы – поле «Код студента».

Таблица 2. Информация об учебных дисциплинах.

Код дисциплины	Название дисциплины	Форма обучения (дневное, вечернее, заочное отделение)
----------------	---------------------	---

Первичный ключ таблицы – поле «Код дисциплины».

Таблица 3. Информация о прохождении учебных дисциплин студентами.

Код студента	Код дисциплины	Посещение занятий по указанной дисциплине (да, нет)	Итоговый результат (баллы, 0-100)
--------------	----------------	---	-----------------------------------

Поля «Код студента» и «Код дисциплины» – составной первичный ключ таблицы. В предположении, что каждый студент может проходить предмет только один раз, пара значений «Код студента» и «Код дисциплины» не может повторяться и однозначно выделяет запись среди всех записей таблицы 3.

При добавлении новых записей в таблицы 1 и 2 уникальные значения первичных ключей должны генерироваться программой автоматически. Пользователь программы не может изменить эти значения ни при каких условиях. Не допускается сохранение записей, в которых осталось незаполненным хотя бы одно поле.

Простейший способ генерации значений первичного ключа – это нумерация записей по порядку.

При добавлении записи в таблицу 3 значения «Код студента» и «Код дисциплины» должны выбираться из таблиц 1 и 2 соответственно. Если такая пара значений уже присутствует в таблице, программа выдает сообщение об этом, и создание новой записи прекращается. Не допускается сохранение записей, в которых осталось незаполненным хотя бы одно поле.

Можно удалять любые записи из таблицы 3.

Записи из таблиц 1 и 2 можно удалять только в случае, если в таблице 3 нет соответствующих значений кодов, то есть если ни одна запись таблицы 3 не ссылается на ту запись, которую пользователь хочет удалить.

## ХОД РЕШЕНИЯ ЗАДАЧИ

1. Создание таблиц (без добавления записей). Индексирование таблиц 1 и 2 по первичным ключам (сразу после создания). Индексирование таблицы 1 по фамилии студента, а таблицы 2 – по названию дисциплины для удобства предъявления записей в последующем.
2. Создание программы заполнения базы данных, состоящей из трех таблиц.
3. Отладка программы и заполнение базы данных.

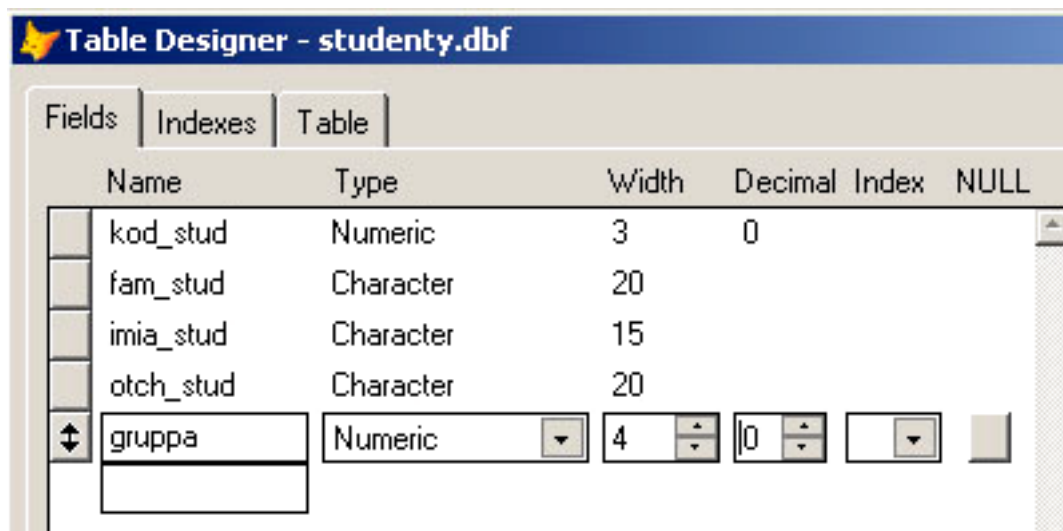
## РЕШЕНИЕ ЗАДАЧИ

### Создание таблиц

Зададим свою рабочую папку.

Назовем таблицу1 *Studenty.dbf* и создадим ее командой  
*create Studenty*

Структура таблицы имеет вид



	Name	Type	Width	Decimal	Index	NULL
	kod_stud	Numeric	3	0		
	fam_stud	Character	20			
	imia_stud	Character	15			
	otch_stud	Character	20			
	gruppa	Numeric	4	0		

Размер поля должен быть таким, чтобы в указанное число позиций помещалось любое предполагаемое значение поля.

В некоторых версиях СУБД FoxPro после создания структуры таблицы предлагается ввести записи. Ответим на этот вопрос отрицательно.

Проиндексируем таблицу по коду студента командой

*index on kod\_stud tag kod\_stud*

Результатом работы команды будет создание структурного мультииндексного файла *Studenty.cdx* с тегом *kod\_stud*.

Проиндексируем таблицу по фамилии студента командой

*index on fam\_stud tag fam\_stud*

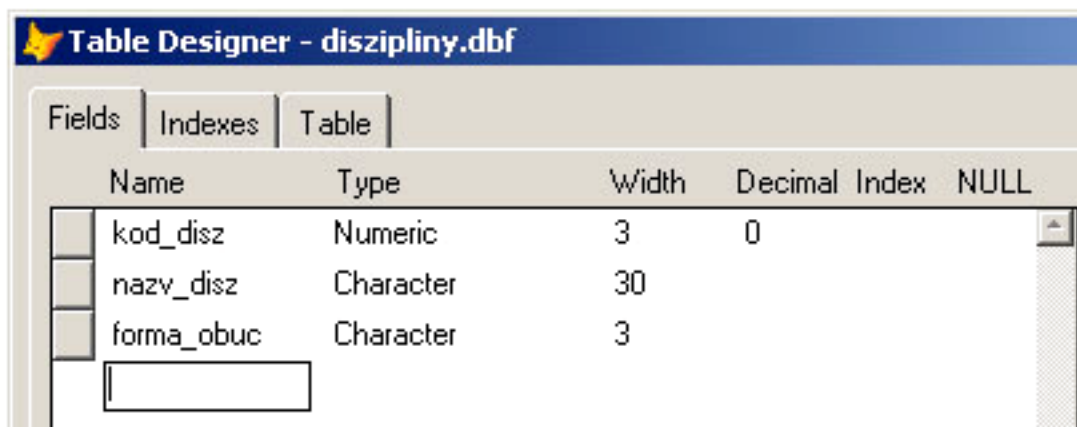
В структурный мультииндексный файл таблицы будет добавлен тег *fam\_stud*.

Конечно, не обязательно, чтобы поле и соответствующий тег назывались одинаково, но одинаковые названия в этом случае допустимы и удобны в использовании.

Назовем таблицу2 *Diszipliny.dbf* и создадим ее командой

*create Diszipliny*

При выполнении этой команды созданная ранее и по умолчанию расположенная в рабочей области *a* таблица *Studenty.dbf* и ее структурный мультииндексный файл будут закрыты, и в область *a* будет помещена вновь создаваемая таблица *Diszipliny.dbf*. Ее структура имеет вид



Name	Type	Width	Decimal	Index	NULL
kod_disz	Numeric	3	0		
nazv_disz	Character	30			
forma_obuc	Character	3			

Поле *forma\_obuc* может принимать значения: *дн*, *веч*, *озо*. Этим определяются тип и размер поля.

Проиндексируем таблицу по коду дисциплины командой

*index on kod\_disz tag kod\_disz*

Результатом работы команды будет создание структурного мультииндексного файла *Diszipliny.cdx* с тегом *kod\_disz*.

Проиндексируем таблицу по названию дисциплины командой

*index on nazv\_disz tag nazv\_disz*

В структурный мультииндексный файл таблицы будет добавлен тег *nazv\_disz*.

Назовем таблицу3 *Resultaty\_po\_Disziplinam.dbf* и создадим ее командой

*create Resultaty\_po\_Disziplinam*

Ее структура имеет вид

Name	Type	Width	Decimal	Index	NULL
kod_stud	Numeric	3	0		
kod_disz	Numeric	3	0		
poseshenie	Logical	1			
itog	Numeric	3	0		

Поле *poseshenie* будет принимать два значения: *true* (.T., да, студент посещал занятия) и *false* (.F., студент не посещал занятия).

Максимальный итоговый балл по результатам изучения курса – 100. В любом случае балл – целое число. Эти два обстоятельства определяют тип и размер поля *itog*.

## Программа

В программе используются следующие переменные:

Имя переменной	Пояснение
n	Выбранный номер действия (добавление, обновление, удаление записи)
m	Номер, соответствующий таблице, в которой нужно произвести изменения
k	Используется для генерации значения первичного ключа при добавлении записей в таблицы 1 и 2
ks	Выбранное значение поля «Код студента» в процедурах для таблицы 3
kd	Выбранное значение поля «Код дисциплины» в процедурах для таблицы 3
nr	Номер выбранной записи в процедуре обновления таблицы 3
nn	Номер, соответствующий части записи, выбранной для обновления в таблице 3

**\*Начало основной программы**

```
close database
set talk off
```

***\*Организация основного меню программы***

```
do while .T.
    clear

    @ 5,1 say 'Выберите действие'
    n=-1
    do while (n<1).or.(n>4)
        @ 6,10 say '1: Добавление записи'
        @ 7,10 say '2: Обновление записи'
        @ 8,10 say '3: Удаление записи'
        @ 9,10 say '4: Выход'
        @ 10,10 get n
        read
    enddo
    if n=4
        return
    endif
    m=-1
    @ 12,1 say 'Выберите таблицу'
    do while (m<1).or.(m>3)
        @ 13,10 say '1 - Информация о студентах'
        @ 14,10 say '2 - Информация об учебных'+;
            'дисциплинах'
        @ 15,10 say '3 - Информация о прохождении'+;
            'учебных дисциплин студентами'
        @ 16,10 get m
        read
    enddo
```

***\*Конец цикла, организующего основное меню***

***\*Вызов процедуры, соответствующей выбранному действию***

```
do case
    case n=1
        do AddRecord
    case n=2
        do UpdateRecord
    case n=3
        do DeleteRecord
endcase
enddo
return
```

**\* Конец основной программы**

*\*Вызов процедуры добавления записей в выбранную таблицу*

### **Procedure AddRecord**

```
do case
  case m=1
    do AddRecord_Stud
  case m=2
    do AddRecord_Disz
  case m=3
    do AddRecord_StudDisz
endcase
```

**return**

*\*Вызов процедуры обновления записей в выбранной таблице*

### **Procedure UpdateRecord**

```
do case
  case m=1
    do UpdateRecord_Stud
  case m=2
    do UpdateRecord_Disz
  case m=3
    do UpdateRecord_StudDisz
endcase
```

**return**

*\*Вызов процедуры удаления записей в выбранной таблице*

### **Procedure DeleteRecord**

```
do case
  case m=1
    do DeleteRecord_Stud
  case m=2
    do DeleteRecord_Disz
  case m=3
    do DeleteRecord_StudDisz
endcase
```

**return**

*\*Процедура добавления записей в таблицу Studenty*

### **Procedure AddRecord\_Stud**

```
clear
use Studenty order tag fam_stud
browse nodelete noedit title 'Добавление записей '+'
      'Ctrl+Y). Выход с сохранением(Ctrl+W)';
fields fam_stud :H='Фамилия студента',;
```



```

        imia_stud :H='Имя студента',;
        otch_stud :H='Отчество студента',;
        группа :H='Номер группы' :V=(группа>0);
                :E='Номер группы - целое '+';
                        'положительное число',;
        kod_stud :H='Код студента' :R
set order to
calc max(kod_stud) to k
go top
do while not eof()
    if kod_stud=0
        if (fam_stud=' ').or.(imia_stud=' ');
                .or.(otch_stud=' ').or.(группа=0)
            delete
        else
            k=k+1
            repl kod_stud with k
        endif
    endif
    skip
enddo
pack
set order to fam_stud
browse noappend noedit nodelete;
    title 'Результат';
    fields fam_stud :H='Фамилия студента',;
            imia_stud :H='Имя студента',;
            otch_stud :H='Отчество студента',;
            группа :H='Номер группы',;
            kod_stud :H='Код студента'
close database

```

**return**

*\*Процедура обновления записей в таблице Studenty*

### **Procedure UpdateRecord\_Stud**

```

clear
use Studenty order tag fam_stud
browse noappend noedit nodelete;
    title 'Выберите запись для редактирования. '+';
            'Выделите запись мышкой, нажмите Esc)';
    fields fam_stud :H='Фамилия студента',;
            imia_stud :H='Имя студента',;
            otch_stud :H='Отчество студента',;
            группа :H='Номер группы',;
            kod_stud :H='Код студента'

```

```

@ 5,1 say 'Редактировать?' get otvet default 'N';
                                function '@M Y,N'
@ 6,1 say 'Выбор ответа - клавиша "пробел" '
read
if otvet='Y'
    k=kod_stud
    set filter to kod_stud=k
    browse noappend nodelete;
        title 'Редактирование записи. Выход с '+'
                                'сохранением (Ctrl+W)';
        fields fam_stud :H='Фамилия студента',;
            imia_stud :H='Имя студента',;
            otch_stud :H='Отчество студента',;
            группа :H='Номер группы' :V=(группа>0);
                                :E='Номер группы - целое '+'
                                'положительное число',;
            kod_stud :H='Код студента' :R
    set filter to
endif

browse noappend noedit nodelete;
    title 'Результат';
    fields fam_stud :H='Фамилия студента',;
        imia_stud :H='Имя студента',;
        otch_stud :H='Отчество студента',;
        группа :H='Номер группы',;
        kod_stud :H='Код студента'

set order to
close database

```

**return**

*\*Процедура удаления записей из таблицы Studenty*

### **Procedure DeleteRecord\_Stud**

```

clear
use Studenty order tag fam_stud
browse noappend noedit nodelete;
    title 'Выберите одну запись для удаления из '+'
        'таблицы (Выделите запись мышкой, нажмите Esc)';
    fields fam_stud :H='Фамилия студента',;
        imia_stud :H='Имя студента',;
        otch_stud :H='Отчество студента',;
        группа :H='Номер группы',;
        kod_stud :H='Код студента'

```

```

@ 5,1 say 'Удалить?' get otvet default 'N';
                                function '@M Y,N'
@ 6,1 say 'Выбор ответа - клавиша "пробел" '
read
if otvet='Y'
    select b
    use Resultaty_po_disziplina
    locate for kod_stud=a.kod_stud
    if found()
        @ 6,1 say 'Выбранную запись удалить'+;
        'нельзя. Она используется в таблице '+;
        'Resultaty_po_disziplina.'
        wait 'Нажмите любую клавишу для просмотра'+;
        ' таблицы и возврата в основное меню.'
    else
        select a
        delete
        pack
    endif
endif
select a
browse noappend noedit nodelete;
    title 'Результат';
    fields fam_stud :H='Фамилия студента',;
           imia_stud :H='Имя студента',;
           otch_stud :H='Отчество студента',;
           grupa :H='Номер группы',;
           kod_stud :H='Код студента'
set order to
close database

```

**return**

*\*Процедура добавления записей в таблицу Diszipliny*

### **Procedure AddRecord\_Disz**

```

clear
use Diszipliny order tag nazv_disz
browse nodelete noedit title 'Добавление записей '+;
    '(Ctrl+Y). Выход с сохранением(Ctrl+W)';
    fields nazv_disz :H='Учебная дисциплина',;
           forma_obuc :H='Форма обучения (дн, веч, озо)',;
           kod_disz :H='Код дисциплины' :R
set order to
calc max(kod_disz) to k
go top

```

```

do while not eof()
  if kod_disz=0
    if (nazv_disz=' ').or.(forma_obuc=' ')
      delete
    else
      k=k+1
      repl kod_disz with k
    endif
  endif
  skip
enddo
pack
set order to nazv_disz
browse noappend noedit nodelete;
  title 'Результат';
  fields nazv_disz :H='Учебная дисциплина',;
  forma_obuc :H='Форма обучения (дн, веч, озо)',;
  kod_disz :H='Код дисциплины'
close database

```

**return**

*\*Процедура обновления записей в таблице Diszipliny*

### **Procedure UpdateRecord\_Disz**

```

clear
use Diszipliny order tag nazv_disz
browse noappend noedit nodelete;
  title 'Выберите запись для редактирования '+'
    '(Выделите запись мышкой, нажмите Esc)';
  fields nazv_disz :H='Учебная дисциплина',;
  forma_obuc :H='Форма обучения (дн, веч, озо)',;
  kod_disz :H='Код дисциплины'
@ 5,1 say 'Редактировать?' get otvet default 'N';
                                function '@M Y,N'
@ 6,1 say 'Выбор ответа - клавиша "пробел" '
read
if otvet='Y'
  k=kod_disz
  set filter to kod_disz=k
  browse noappend nodelete;
    title 'Редактирование записи. Выход с '+'
      'сохранением (Ctrl+W)';
  fields nazv_disz :H='Учебная дисциплина',;
  forma_obuc :H='Форма обучения (дн, веч, озо)',;
  kod_disz :H='Код дисциплины' :R

```

```

        set filter to

endif
browse noappend noedit nodelete;
    title 'Результат';
    fields nazv_disz :H='Учебная дисциплина',;
    forma_obuc :H='Форма обучения (дн, веч, озо)',;
    kod_disz :H='Код дисциплины'
set order to
close database
return

*Процедура удаления записей из таблицы Diszipliny
Procedure DeleteRecord_Disz
    clear
    use Diszipliny order tag nazv_disz
    browse noappend noedit nodelete;
        title 'Выберите одну запись для удаления из '+'
            'таблицы (Выделите запись мышкой, нажмите Esc)';
        fields nazv_disz :H='Учебная дисциплина',;
        forma_obuc :H='Форма обучения (дн, веч, озо)',;
        kod_disz :H='Код дисциплины'
    @ 5,1 say 'Удалить?' get otvet default 'N';
                                function '@M Y,N'
    @ 6,1 say 'Выбор ответа - клавиша "пробел" '
    read
    if otvet='Y'
        select b
        use Resultaty_po_disziplinam
        locate for kod_disz=a.kod_disz
        if found()
            @ 6,1 say 'Выбранную запись удалить'+;
                'нельзя. Она используется в таблице'+;
                ' Resultaty_po_disziplinam.'
            wait 'Нажмите любую клавишу для просмотра'+;
                ' таблицы и возврата в основное меню.'
        else
            select a
            delete
            pack
        endif
    endif
    select a
    browse noappend noedit nodelete;
        title 'Результат';

```

```

        fields nazv_disz :H='Учебная дисциплина',;
        forma_obuc :H='Форма обучения (дн, веч, озо)',;
        kod_disz :H='Код дисциплины'
    set order to
    close database

```

**return**

*\*Процедура добавления записей в таблицу*

*\*Resultaty\_po\_Disziplinan*

### **Procedure AddRecord\_StudDisz**

```

    clear
    use Studenty order tag fam_stud
    browse noappend noedit nodelete;
        title 'Выберите студента '+';
            '(Выделите запись мышкой, нажмите Esc)';
        fields fam_stud :H='Фамилия студента',;
            imia_stud :H='Имя студента',;
            otch_stud :H='Отчество студента',;
            gruppa :H='Номер группы',;
            kod_stud :H='Код студента'
    ks=kod_stud
    select b
    use Diszipliny order tag nazv_disz
    browse noappend noedit nodelete;
        title 'Выберите учебную дисциплину '+';
            '(Выделите запись мышкой, нажмите Esc)';
        fields nazv_disz :H='Учебная дисциплина',;
            forma_obuc :H='Форма обучения (дн, веч, озо)',;
            kod_disz :H='Код дисциплины'
    kd=kod_disz
    select c
    use Resultaty_po_disziplinam
    locate for (kod_stud=ks).and.(kod_disz=kd)
    if found()
        @ 6,1 say 'Такая запись уже есть.'+;
            'Ее можно обновить.'
        wait 'Нажмите любую клавишу для возврата'+;
            'в основное меню.'
        close database
        return
    endif
    go bottom
    append blank
    repl kod_stud with ks
    repl kod_disz with kd

```

```

select a
set order to kod_stud
select b
set order to kod_disz
select c
set relation to kod_stud into a
set relation to kod_disz into b additive
set filter to (kod_stud=ks).and.(kod_disz=kd)
browse noappend nodelete;
    title 'Введите недостающую информацию. '+'
        'Выход с сохранением (Ctrl+W)';
    fields kod_stud :H='Код студента' :R,;
        a.fam_stud :H='Фамилия студента' :R,;
        kod_disz :H='Код дисциплины' :R,;
        b.nazv_disz :H='Учебная дисциплина' :R,;
        poseshenie :H='Посещение занятий',;
        itog :H='Итоговые баллы по дисциплине';
            :V=(itog>=0);
            :E='Итоговые баллы - целое '+'
                'неотрицательное число'
if IsBlank(poseshenie).or.IsBlank(itog)
    delete
endif
set filter to
pack
set relation to kod_stud into a
set relation to kod_disz into b additive
browse noappend noedit nodelete;
    title 'Результат';
    fields kod_stud :H='Код студента',;
        a.fam_stud :H='Фамилия студента',;
        kod_disz :H='Код дисциплины',;
        b.nazv_disz :H='Учебная дисциплина',;
        poseshenie :H='Посещение занятий',;
        itog :H='Итоговые баллы по дисциплине'
set relation to
close database

```

**return**

*\*Процедура обновления записей в таблице*

*\*Resultaty\_po\_Disziplinan*

### **Procedure UpdateRecord\_StudDisz**

```

clear
use Studenty order tag kod_stud
select b

```

```

use Diszipliny order tag kod_disz
select c
use Resultaty_po_disziplinam
set relation to kod_stud into a
set relation to kod_disz into b additive
browse noappend noedit nodelete;
    title 'Выберите запись для изменения '+';
      '(Выделите запись мышкой, нажмите Esc)';
    fields kod_stud :H='Код студента',;
      a.fam_stud :H='Фамилия студента',;
      kod_disz :H='Код дисциплины',;
      b.nazv_disz :H='Учебная дисциплина',;
      poseshenie :H='Посещение занятий',;
      itog :H='Итоговые баллы по дисциплине'
ks=kod_stud
kd=kod_disz
nr=recno()
set relation to
nn=-1
@ 5,10 say 'Выберите действие'
do while (nn<1).or.(nn>3)
    @ 6,10 say '1: Изменить фамилию студента '+';
      'в выбранной записи'
    @ 7,10 say '2: Изменить название дисциплины '+';
      'в выбранной записи'
    @ 8,10 say '3: Изменить посещение или баллы '+';
      'в выбранной записи'
    @ 10,10 get nn
    read
enddo
do case
    case nn=1
        select a
        set order to fam_stud
        browse noappend noedit nodelete;
            title 'Выберите студента (Выделите '+';
              'запись мышкой, нажмите Esc)';
            fields fam_stud :H='Фамилия студента',;
              imia_stud :H='Имя студента',;
              otch_stud :H='Отчество студента',;
              grupa :H='Номер группы',;
              kod_stud :H='Код студента'
        select c
        locate for (kod_stud=a.kod_stud);
          .and.(kod_disz=kd)

```



```

if found()
    @ 12,10 say 'Такая запись уже есть. '+';
                'Ее можно обновить.'
    wait 'Нажмите любую клавишу '+';
        'для возврата в основное меню.'
    close database
    return
endif
go nr
repl kod_stud with a.kod_stud
select a
set order to kod_stud
select c
set relation to kod_stud into a
set relation to kod_disz into b additive
browse noappend noedit nodelete;
    title 'Результат';
    fields kod_stud :H='Код студента',;
        a.fam_stud :H='Фамилия студента',;
        kod_disz :H='Код дисциплины',;
        b.nazv_disz :H='Учебная дисциплина',;
        poseshenie :H='Посещение занятий',;
        itog :H='Итоговые баллы по дисциплине'

case nn=2
    select b
    set order to nazv_disz
    browse noappend noedit nodelete;
    title 'Выберите дисциплину (Выделите '+';
        'запись мышкой, нажмите Esc)';
    fields nazv_disz :H='Учебная дисциплина',;
        forma_obuc :H='Форма обучения '+';
                    '(дн, веч, озо)',;
        kod_disz :H='Код дисциплины'
    select c
    locate for (kod_stud=ks);
                .and.(kod_disz=b.kod_disz)
    if found()
        @ 12,10 say 'Такая запись уже есть. '+';
                    'Ее можно обновить.'
        wait 'Нажмите любую клавишу '+';
            'для возврата в основное меню.'
        close database
        return

```

```

endif
go nr
repl kod_disz with b.kod_disz
select b
set order to kod_disz
select c
set relation to kod_stud into a
set relation to kod_disz into b additive
browse noappend noedit nodelete;
    title 'Результат';
    fields kod_stud :H='Код студента',;
        a.fam_stud :H='Фамилия студента',;
        kod_disz :H='Код дисциплины',;
        b.nazv_disz :H='Учебная дисциплина',;
        poseshenie :H='Посещение занятий',;
        itog :H='Итоговые баллы по дисциплине'
case nn=3
set relation to kod_stud into a
set relation to kod_disz into b additive
set filter to (kod_stud=ks);
                .and.(kod_disz=kd)
browse noappend nodelete;
    title 'Внесите изменения. '+';
        'Выход с сохранением (Ctrl+W)';
    fields kod_stud :H='Код студента' :R,;
    a.fam_stud :H='Фамилия студента' :R,;
    kod_disz :H='Код дисциплины' :R,;
    b.nazv_disz :H='Учебная дисциплина' :R,;
    poseshenie :H='Посещение занятий',;
    itog :H='Итоговые баллы по дисциплине'
browse noappend noedit nodelete;
    title 'Вид записи после редактирования';
    fields kod_stud :H='Код студента',;
        a.fam_stud :H='Фамилия студента',;
        kod_disz :H='Код дисциплины',;
        b.nazv_disz :H='Учебная дисциплина',;
        poseshenie :H='Посещение занятий',;
        itog :H='Итоговые баллы по дисциплине'
set filter to
browse noappend noedit nodelete;
    title 'Результат';
    fields kod_stud :H='Код студента',;
        a.fam_stud :H='Фамилия студента',;
        kod_disz :H='Код дисциплины',;
        b.nazv_disz :H='Учебная дисциплина',;

```

```

        poseshenie :H='Посещение занятий',;
        itog :H='Итоговые баллы по дисциплине'
    endcase
    set relation to
    close database

```

**return**

*\*Процедура удаления записей из таблицы*

*\*Resultaty\_po\_Diszipliman*

### **Procedure DeleteRecord\_StudDisz**

```

    clear
    use Studenty order tag kod_stud
    select b
    use Diszipliny order tag kod_disz
    select c
    use Resultaty_po_disziplinam
    set relation to kod_stud into a
    set relation to kod_disz into b additive
    browse noappend noedit;
        title 'Отметьте записи, которые следует удалить'+;
        ' (Ctrl+T). Выход с сохранением (Ctrl+W)';
        fields kod_stud :H='Код студента',;
            a.fam_stud :H='Фамилия студента',;
            kod_disz :H='Код дисциплины',;
            b.nazv_disz :H='Учебная дисциплина',;
            poseshenie :H='Посещение занятий',;
            itog :H='Итоговые баллы по дисциплине'
    pack
    set relation to kod_stud into a
    set relation to kod_disz into b additive
    browse noappend noedit nodelete;
        title 'Результат';
        fields kod_stud :H='Код студента',;
            a.fam_stud :H='Фамилия студента',;
            kod_disz :H='Код дисциплины',;
            b.nazv_disz :H='Учебная дисциплина',;
            poseshenie :H='Посещение занятий',;
            itog :H='Итоговые баллы по дисциплине'
    set relation to
    close database
    return

```

## СОДЕРЖАНИЕ

Постановка задачи.....	3
Ход решения задачи.....	4
Решение задачи	
– Создание таблиц.....	4
– Программа.....	6